Generalization of automatic sequences for numeration systems on a regular language

Michel Rigo
Institut de Mathématiques, Université de Liège,
Grande Traverse 12 (B 37), B-4000 Liège, Belgium.
M.Rigo@ulg.ac.be

June 21, 1999

Abstract

Let L be an infinite regular language on a totally ordered alphabet $(\Sigma, <)$. Feeding a finite deterministic automaton (with output) with the words of L enumerated lexicographically with respect to $<$ leads to an infinite sequence over the output alphabet of the automaton. This process generalizes the concept of k-automatic sequence for abstract numeration systems on a regular language (instead of systems in base k). Here, I study the first properties of these sequences and their relations with numeration systems.

1 Introduction

In [9], P. Lecomte and I have defined a *numeration system* as being a triple $S = (L, \Sigma, <)$ where L is an infinite regular language over a totally ordered alphabet $(\Sigma, <)$. The lexicographic ordering of L gives a one-to-one correspondence r_S between the set of the natural numbers $\mathbb{N}$ and the language L.

For a given subset X of $\mathbb{N}$, a question arise naturally. Is it possible to find a numeration system S such that $\mathrm{r}_S(X)$ is recognizable by finite automata ? (In this case, X is said to be *S-recognizable*.) For example, the set $\{n^2 : n \in \mathbb{N}\}$ is S-recognizable for some S and the arithmetic progressions $p + q\mathbb{N}$ are S-recognizable for any S. An interesting question is thus the following: is there a system S such that the set of primes is S-recognizable ?

To answer this question I show that a subset of $\mathbb{N}$ is S-recognizable if and only if its characteristic sequence can be generated by an 'automatic' method. The term automatic refers, as we shall see further, to a generalization of the k-automatic sequences for numeration systems on a regular language.

The k-automatic sequences are well-known and have been extensively studied since the 70's [2, 5, 7, 14]. The construction of this kind of sequences is based on the representation of the integers in the base k. For a given integer n, one represents this number in base k using the greedy algorithm and obtains a word $[n]_k$ over the alphabet $\{0, \ldots, k-1\}$. Next one gives $[n]_k$ to a deterministic finite automaton with output and obtains the n^{th} term of a sequence which is said to be a k-automatic sequence.

These sequences have been already generalized in different ways [2]. In particular, a method used by J. Shallit to generalize the k-automatic sequences is to consider some kind of linear numeration system instead of the standard numeration system with integer base k [14]. Two properties of the systems encountered in [14] are precisely that the set of all the representations is regular and that the lexicographic ordering is respected.

Here, instead of giving $[n]_k$ to a deterministic finite automaton with output, we feed it with $\mathrm{r}_S(n)$ to obtain an output which is the n^{th} term of an S*-automatic sequence* for a numeration system S. Having thus introduced the concept of S-automatic sequences, we can follow two paths. Learn their intrinsic properties but also use them as a tool to check if a subset of $\mathbb{N}$ is S-recognizable.

Our article has the following articulation. In the first section, we recall some definitions and we introduce a teaching example which could be very instructive for the reader not familiar with automatic sequences. In the second section, we adapt the classical results concerning the fiber and the kernel of an automatic sequence.

Initially, A. Cobham showed the equivalence between the k-automatic sequences and the sequences obtained by iterating a uniform morphism (also called uniform tag system [7]). In the third section, we show that an S-automatic sequence is always generated by a substitution (i.e., an iterated non-uniform morphism followed by one application of another morphism). From this, we deduce that the number of distinct factors of length l in an S-automatic sequence is in $O(l^2)$. We also show how to construct S-automatic sequences with at least the same complexity that infinite words obtained by iterated morphisms.

In the last section, we will be able to show that for any numeration system S, the set of primes is never S-recognizable. We use the fact that to be S-recognizable, the characteristic sequence of the set must be generated by a substitution. Hence we use some results of C. Mauduit about the density of the infinite words obtained by substitution [11, 12].

2 Basic definitions and notations

In this paper, capital greek letters represent finite alphabet. We denote by Σ^* the set of the words over Σ (ε is the empty word) and by Σ^ω the set of the infinite words over Σ. If K is a set then $\#K$ denotes the cardinality of K and if w is a string then $|w|$ denotes the length of w. For $1 \le i \le |w|$, w_i is the i^{th} letter of w. The same notation holds for infinite words, in this case $i \in \mathbb{N} \setminus \{0\}$.

First, recall some definitions about the numeration systems we are dealing with. For more about these systems see [9].

Definition 1 A *numeration system* S is a triple $(L, \Sigma, <)$ where L is an infinite regular language over the totally ordered alphabet $(\Sigma, <)$.

For each $n \in \mathbb{N}$, $\mathrm{r}_S(n)$ denotes the $(n+1)^{th}$ word of L with respect to the lexicographic ordering and is called the S*-representation* of n.

Remark that the map $\mathrm{r}_S : \mathbb{N} \to L$ is an increasing bijection. For $w \in L$, we set $\mathrm{val}_S(w) = \mathrm{r}_S^{-1}(w)$. We call $\mathrm{val}_S(w)$ the *numerical value* of w.

Examples of such systems are the numeration systems defined by a recurrence relation whose characteristic polynomial is the minimum polynomial of a Pisot number [3]. (Indeed, with this hypothesis, the set of representations of the

integers is a regular language.) The standard numeration systems with integer base and also the Fibonacci system belong to this class.

Definition 2 Let S be a numeration system. A subset X of $\mathbb{N}$ is *S-recognizable* if $\mathrm{r}_S(X)$ is recognizable by finite automata.

Let us introduce the concept of S-automatic sequence which naturally generalizes the k-automatic sequences based on the representation of the integers in base k. For more about k-automatic sequences see for instance [2, 5].

Definition 3 A *deterministic finite automaton with output* (DFAO) M is a 6-uple $(K, s, \Sigma, \delta, \Delta, \tau)$ where K is the finite set of the states, s is the start state, Σ is the input alphabet, $\delta : K \times \Sigma \to K$ is the transition function, Δ is the output alphabet and $\tau : K \to \Delta$ is the output function.

Definition 4 Let $S = (L, \Sigma, <)$ be a numeration system. A sequence $u \in \Delta^\omega$ is *S-automatic* if there exists a DFAO $M = (K, s, \Sigma, \delta, \Delta, \tau)$ such that for all $n \in \mathbb{N}$,

$$u_{n+1} = \tau(\delta(s, \mathrm{r}_S(n))).$$

If the context is clear, we write $\tau(w)$ in place of $\tau(\delta(s, w))$.

Remark 1 A subset $X \subset \mathbb{N}$ is S-recognizable if and only if its characteristic sequence $\chi_X \in \{0, 1\}^\omega$ is S-automatic.

In the following we will often encounter two more 'classical' ways of obtaining infinite sequences.

Definition 5 Let $\varphi : \Sigma \to \Sigma^*$ be a morphism of monoid such that for some $\sigma \in \Sigma$, $\varphi(\sigma) \in \sigma\Sigma^*$. The word $u_\varphi = \varphi^\omega(\sigma)$ is a fixed point of φ and we say that u_φ is *generated by an iterated morphism.*

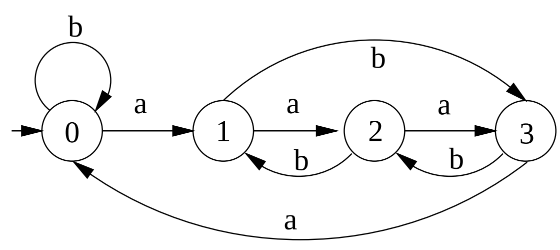
A morphism is *uniform* if $|\varphi(\sigma_1)| = \ldots = |\varphi(\sigma_n)|$, $\Sigma = \{\sigma_1, \ldots, \sigma_n\}$.

Definition 6 A *substitution* T is a triple (φ, h, c) such that $\varphi : \Sigma \to \Sigma^*$ and $h : \Sigma \to \Delta^*$ are morphisms of monoids. Moreover $c \in \Sigma$, $\varphi(c) \in c\Sigma^*$ and for any $\sigma \in \Sigma$, $h(\sigma) = \varepsilon$ or $h(\sigma) \in \Delta$ (h is said to be a *weak coding*). We said that the word $u_T = h(\varphi^\omega(c))$ over Δ is *generated by the substitution T*.

If $h(\sigma) = \varepsilon$ for some σ then h is said to be *erasing* otherwise h is said to be *non-erasing.*

2.1 A teaching example

We consider the numeration system $S = (a^*b^*, \{a, b\}, a < b)$, the alphabets $\Sigma = \{a, b\}$, $\Delta = \{0, 1, 2, 3\}$ and the following DFAO



As usual the start state is indicated by an unlabeled arrow. The first words of a^*b^* are

$$\varepsilon, a, b, aa, ab, bb, aaa, aab, abb, bbb, \ldots$$

and thus feeding the automaton with these words we obtain the first terms of the sequence $u \in \Delta^\omega$,

$$u = 0102303120023101012302303120312023100231012301012 3\ldots.$$

Remark 2 The sequence u is not ultimately periodic. One can observe that the distance between two occurrences of the block '00' is not bounded. Indeed,

$$\tau(w) = 0 \Leftrightarrow \exists r, s \in \mathbb{N} : w = a^{4r}b^s \tag{1}$$

thus a block '00' comes from two consecutive words b^{4r-1} and a^{4r}, $r \geq 1$ and the number of words of length n in a^*b^* is $n+1$, $n \in \mathbb{N}$.

Remark 3 The sequence u is not generated by an iterated morphism φ. First observe that

$$\tau(w) = \left\{ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \right\} \Leftrightarrow \exists r, s \in \mathbb{N} : w = \left\{ \begin{array}{l} a^{4r+1}b^{3s},\ a^{4r+2}b^{3s+1},\ a^{4r+3}b^{3s+2} \\ a^{4r+1}b^{3s+2},\ a^{4r+2}b^{3s},\ a^{4r+3}b^{3s+1} \\ a^{4r+1}b^{3s+1},\ a^{4r+2}b^{3s+2},\ a^{4r+3}b^{3s}. \end{array} \right.$$

Suppose that there exists a morphism φ such that $u = \lim_{n\to+\infty} \varphi^n(0)$.

1) If $\varphi(0) \in 0102\Delta^*$ then the block '0102' must appear at least twice in u since '0' appears twice in u. If the first '0' of the block is obtained from a word $a^{4r}b^s$ with $r \geq 1$ then the second '0' is obtained from $a^{4r-2}b^{s+2}$ which leads to a contradiction in view of (1). If the first '0' is obtained from b^s with $s \geq 1$ then the second '0' come from a^sb and we have $s = 4t$. The '2' is obtained from $a^{4t-1}b^2$ which also leads to a contradiction.

2) If $\varphi(0) = 01$ then in view of the first terms of u, $\varphi(1) \in 023031200231\Delta^*$. We show that '023031200' appears only once in u. Suppose that we can find another block of this kind. Thus the last two '0' come from words b^{4r-1} and a^{4r} with $r \geq 2$. Since we consider all the words of a^*b^* lexicographically ordered, the first '0' of the block come from a^7b^{4r-8} which is in contradiction with (1).

3) If $\varphi(0) = 010$ then $\varphi(1) \in 23031200231\Delta^*$ and $\varphi(010) \in 01023031200\Delta^*$. The block '010' appears at least twice in u but we know that '023031200' appears only once.

We shall see further that u is generated by a substitution.

3 First results about S-automatic sequences

Some classical results about k-automatic sequences can be easily restated [7, 5].

Definition 7 Let $a \in \Delta$ and $S = (L, \Sigma, <)$, the *S-fiber* $\mathcal{F}_S(u, a)$ of a sequence $u \in \Delta^\omega$ is defined as follows

$$\mathcal{F}_S(u, a) = \{\mathrm{r}_S(n) \,:\, u_n = a\}.$$

Theorem 8 *Let u be an infinite sequence over Δ and $S = (L, \Sigma, <)$. The sequence u is S-automatic if and only if for all $a \in \Delta$, $\mathcal{F}_S(u, a)$ is a regular subset of L.*

Proof. If u is S-automatic then we have a DFAO $M = (K, s, \Sigma, \delta, \Delta, \tau)$ which is used to generate u. Let $L(M')$ be the language recognized by the DFA $M' = (K, s, \Sigma, \delta, F)$ where the set of final states F only contains the states k such that $\tau(k) = a$. Therefore $\mathcal{F}_S(u, a)$ is regular since it is the intersection of the two regular sets $L(M')$ and L.

The condition is sufficient. Let $\Delta = \{a_1, \ldots, a_n\}$. Remark that if $i \neq j$, $\mathcal{F}_S(u, a_i) \cap \mathcal{F}_S(u, a_j) = \emptyset$ and $L = \cup_{i=1}^n \mathcal{F}_S(u, a_i)$. For all $i = 1, \ldots, n$, $\mathcal{F}_S(u, a_i)$ is accepted by a DFA $M_i = (K_i, s_i, \Sigma, \delta_i, F_i)$. From these automata we construct a DFAO $M = (K, s, \Sigma, \delta, \Delta, \tau)$ to generate u using the numeration system S. The set K is $K_1 \times \ldots \times K_n$, the initial state is $(s_1, \ldots, s_n)$. For all states $(q_1, \ldots, q_n) \in K$ and for all $\sigma \in \Sigma$, $\delta((q_1, \ldots, q_n), \sigma) = (\delta_1(q_1, \sigma), \ldots, \delta_n(q_n, \sigma))$. If there is a unique i such that $q_i \in F_i$ then $\tau((q_1, \ldots, q_n)) = a_i$ otherwise the state cannot be reached by a word of L and the output is not important. The sequence u is obtained from S and the DFAO M thus u is S-automatic. □

The notion of k-kernel of a k-automatic sequence can be transposed as follows.

Definition 9 Let $S = (L, \Sigma, <)$ and u be an infinite sequence. For each $w \in \Sigma^*$, we set $\mathcal{K}_w = \{v \in L \,|\, \exists z \in \Sigma^* : v = wz\}$. One can enumerate $\mathcal{K}_w$ lexicographically with respect to $<$, $\mathcal{K}_w = \{wz_0 < wz_1 < \ldots\}$. Thus for each $w \in \Sigma^*$, one can construct the subsequence $n \mapsto u_{\mathrm{val}_S(wz_n)}$ (remark that the subsequence can be finite or even empty).

Theorem 10 *Let $S = (L, \Sigma, <)$. A sequence $u \in \Delta^\omega$ is S-automatic if and only if $\{n \mapsto u_{\mathrm{val}_S(wz_n)} : w \in \Sigma^*\}$ is finite.*

Proof. If u is S-automatic, we have a DFAO $M = (K, s, \Sigma, \delta, \Delta, \tau)$ used to generate u and we define the equivalence relation $\sim_1$ over Σ^* by $x \sim_1 y$ if and only if $\delta(s, x) = \delta(s, y)$. In the same way, the minimal automaton of L provides an equivalence relation $\sim_2$. The two relations have a finite index thus the relation $\sim_{1,2}$ given by $x \sim_{1,2} y$ if and only if $x \sim_1 y$ and $x \sim_2 y$ has also a finite index. Remark that each class of $\sim_{1,2}$ gives one of the sequences $n \mapsto u_{\mathrm{val}_S(wz_n)}$. Indeed, $x \sim_2 y$ implies that $\{z \in \Sigma^* : xz \in L\} = \{z \in \Sigma^* : yz \in L\}$ thus $\mathcal{K}_x = \{xz_0 < xz_1 < \ldots\}$ and $\mathcal{K}_y = \{yz_0 < yz_1 < \ldots\}$ with the same $z_0, z_1, \ldots$.

The condition is sufficient. We show how to construct a DFAO. The states are the subsequences $q_w = (n \mapsto u_{\mathrm{val}_S(wz_n)})$. The initial state is q_ε (i.e., the subsequence obtained from the empty word). The transition function δ is given by $\delta(q_w, \sigma) = q_{w\sigma}$ and the output function τ is given by $\tau(q_w) = u_{\mathrm{val}_S(w)}$. □

4 Complexity of S-automatic sequences

The complexity function p_u of an infinite sequence u maps $n \in \mathbb{N}$ to the number $p_u(n)$ of distinct factors of length n which occur at least once in u. In this section, we will show that the complexity of an S-automatic sequence is in $O(n^2)$ as a consequence that every S-automatic sequence is generated by a substitution.

Recall that an infinite word w generated by iterated morphism has a complexity such that

$$c_1 f(n) \leq p_w(n) \leq c_2 f(n)$$

where $f(n)$ is one of the following functions 1, n, $n \log\log n$, $n \log n$ or n^2 [13]. For a survey on the complexity function, see for example [1].

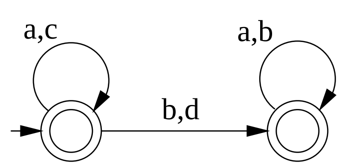
The next remark shows that an S-automatic sequence can reach at least the same complexity as a word generated by morphism.

Remark 4 For every infinite word w generated by an iterated morphism φ over an alphabet Δ we can construct an S-automatic sequence u such that $\forall n \in \mathbb{N}$, $p_w(n) \leq p_u(n)$.

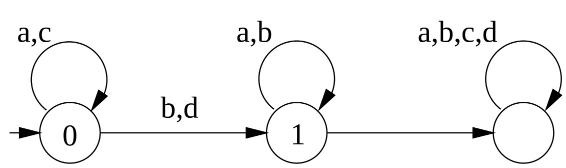
We show how to proceed on the following example,

$$\Delta = \{0,1\}, \ \varphi : \left\{ \begin{array}{l} 0 \mapsto 0101 \\ 1 \mapsto 11. \end{array} \right.$$

It is well-known that $w = \varphi^\omega(0)$ is such that p_w is of complexity $O(n \log\log n)$ [13]. To the morphism φ, we associate a finite automaton M (if the morphism is not uniform then M is not deterministic). The set of states is Δ, all the states are final and the transition function δ is obtained by reading the productions of φ from left to right. For this purpose, we introduce a new ordered alphabet Σ such that $\#\Sigma = \sup_{x\in\Delta} |\varphi(x)|$. Here, 0 gives the initial state (for we consider the word $\varphi^\omega(0)$) and 1 the other state. Thus with $\Sigma = \{a < b < c < d\}$, we have $\delta(0,a) = [\varphi(0)]_1 = 0$, $\delta(0,b) = [\varphi(0)]_2 = 1, \ldots$ and M is then



As is customary, the final states are denoted by double circles. The language accepted by M is $L = \{a,c\}^*\{b,d\}\{a,b\}^* \cup \{a,c\}^*$. The numeration system S is thus $(L, \Sigma, a < b < c < d)$. This kind of construction can also be found in [10]. Now from M we simply construct a DFAO M'



The way we find the output can be easily understood. The third state can have any output for this state is never reached with a word belonging to L. One remarks that the S-automatic sequence obtained with M' and S is

$$u = \varphi(0)\varphi^2(0)\varphi^3(0)\ldots$$

and thus every factor of $w = \varphi^\omega(0)$ belongs to u.

We now show that every S-automatic sequence is generated by a substitution.

Lemma 11 *Let $\Sigma = \{\sigma_1 < \ldots < \sigma_n\}$, $M = (K, s, \Sigma, \delta, F)$ be a DFA and $\alpha \notin K$. The morphism $\varphi_M : K \cup \{\alpha\} \to (K \cup \{\alpha\})^*$ defined by*

$$\begin{cases} \alpha \mapsto \alpha s \\ k \mapsto \delta(k, \sigma_1) \ldots \delta(k, \sigma_n), \; k \in K \end{cases}$$

produces the sequence u_φ of the states reached by the words of Σ^* i.e., $\forall i \in \mathbb{N} \setminus \{0\}$, $u_{i+1} = \delta(s, w_i)$ where w_i is the i^{th} element of $(\Sigma^*, <)$.

Proof. One can check easily by constructing $\varphi(\alpha)$, $\varphi^2(\alpha)$, $\varphi^3(\alpha)$ (which are prefixes of u_φ) that u_φ satisfies the property. $\square$

Proposition 12 *Every S-automatic sequence is generated by a substitution.*

Proof. Let $S = (L, \Sigma, <)$, $M_L = (K, s, \Sigma, \delta, F)$ be a DFA accepting L and u be an S-automatic sequence obtained with the DFAO $\mathcal{M} = (K', s', \Sigma, \delta', \Delta, \tau)$. From these two automata, we construct the product automaton $M = (K \times K', (s, s'), \Sigma, \nu)$ where $\nu((k, k'), \sigma) = (\delta(k, \sigma), \delta'(k', \sigma))$. We do not give explicitly the final states of M. By Lemma 11, we associate to this automaton a morphism $\varphi_M : (K \times K') \cup \{\alpha\} \to ((K \times K') \cup \{\alpha\})^*$. To conclude the proof, we construct the erasing morphism $h : (K \times K') \cup \{\alpha\} \to \Delta^*$ defined by

$$\begin{cases} \quad h(\alpha) & = \varepsilon & \\ h((k, k')) & = \varepsilon & \text{if } k \notin F \\ & = \tau(k') & \text{otherwise.} \end{cases}$$

Indeed, $\varphi^\omega(\alpha)$ is the sequence of the states reached by the words of Σ^* in M but we are only interested in the words belonging to L and in the corresponding output of $\mathcal{M}$. Thus u is generated by (φ_M, h, α). $\square$

Dealing with erasing morphisms whenever one wants to determine the complexity function of a sequence is painful. So the next lemma permits to get rid of erasing morphisms.

Lemma 13 [6] *If f and g are arbitrary morphisms with $f(g^\omega(a))$ an infinite word, then there exists a non-erasing morphism k and a coding h (i.e., a letter-to-letter morphism h) such that $f(g^\omega(a)) = h(k^\omega(a))$.* $\square$

Theorem 14 *The complexity of an automatic sequence is in $O(n^2)$. Moreover, there exists an automatic sequence v and a positive constant d' such that $\forall n > 0$, $p_v(n) \geq d'n^2$.*

Proof. Let u be an S-automatic sequence. By Proposition 12, u is generated by a substitution (φ, h, α) and by Lemma 13 we can suppose that h is non-erasing. The word $u_\varphi = \varphi^\omega(\alpha)$ is generated by an iterated morphism and thus $p_{u_\varphi}(n) \leq d\, n^2$. To conclude, since $u = h(u_\varphi)$, recall that if v, w are two infinite words and if h is a non-erasing morphism such that $h(v) = w$ then there exist positive constants a, b such that $p_w(n) \leq a\, p_v(n + b)$ [13].

2) We show that there exist a language L over an ordered alphabet and a DFAO such that the corresponding automatic sequence v has a complexity function $p_v(n) \geq d'n^2$.

The morphism

$$\varphi : \begin{cases} 0 \mapsto 01 \\ 1 \mapsto 12 \\ 2 \mapsto 2 \end{cases}$$

generates the word $w = \varphi^\omega(0)$. Since 2 is a bounded letter (i.e., $|\varphi^n(2)|$ is bounded) and 2^n is a factor of w for an arbitrary n, there exists a positive constant d' such that $p_w(n) \geq d'\, n^2$ (see [13]). Using the same technique as in Remark 4, we construct an S-automatic sequence v such that $p_v(n) \geq p_w(n)$. One find easily that the regular language used in the numeration system S is $L = a^* \cup a^*ba^* \cup a^*ba^*ba^*$. $\square$

To conclude this section, we refine in a very simple way Proposition 12 to give a characterization of the S-automatic sequences.

Let $T = (\varphi, h, c)$ and $T' = (\varphi', h', c')$ be two substitutions such that $\varphi : \Sigma \to \Sigma^*$, $h : \Sigma \to \Delta^*$,$\varphi' : \Sigma' \to \Sigma'^*$ and $h' : \Sigma' \to \Delta'^*$. A morphism of substitutions $m : T \to T'$ is a surjective morphism $m : \Sigma \cup \Delta \to \Sigma' \cup \Delta'$ such that

1. $m(c) = c'$, $m(\Sigma) = \Sigma'$, $m(\Delta) = \Delta'$
2. $m(\varphi(\sigma)) = \varphi'(m(\sigma))$, $\forall \sigma \in \Sigma$
3. $m(h(\sigma)) = h'(m(\sigma))$, $\forall \sigma \in \Sigma$.

For a regular language L on the totally ordered alphabet $(\Sigma, <)$ and for a DFAO $M = (K, s, \Sigma, \delta, \Delta, \tau)$, one can construct the *canonical substitution* $T_{(L,<,M)}$ by proceeding in the same way as in Proposition 12 with M_L equals to the minimal automaton of L and the DFAO $\mathcal{M}$ equals to a reduced and accessible copy of M.

To reduce M, one have to merge the states p, q such that for all $w \in \Sigma^*$, $\tau(\delta(p, w)) = \tau(\delta(q, w))$.

Definition 15 A substitution T is an $(L, <, M)$-*substitution* if there exists a morphism $m : T \to T_{(L,<,M)}$. This kind of construction has already been introduced in [3] for linear numeration systems based on a Pisot number.

The next theorem is obvious and we state it without proof.

Theorem 16 *Let $S = (L, \Sigma, <)$. The sequence $u \in \Delta^\omega$ is S-automatic if and only if u is generated by a $(L, <, M)$-substitution for some DFAO M.* $\square$

5 Application to S-recognizable sets of integers

Proposition 12 gives a necessary condition for a set X of integers to be S-recognizable. The characteristic sequence $\chi_X \in \{0, 1\}^\omega$ has to be generated by a substitution. Thus this proposition can be used as an interesting tool to show that a subset of $\mathbb{N}$ is not S-recognizable for any numeration system S.

In the following $\mathcal{P}$ is the set of primes and $\chi_\mathcal{P}$ is its characteristic sequence. We show that $\mathcal{P}$ is never S-recognizable but first we construct by hand a subset of $\mathbb{N}$ which cannot be S-recognizable for its characteristic sequence is too complex.

Example 1 For $n \geq 3$, consider the $\binom{n}{3}$ words belonging to $\{0,1\}^n$ which contains exactly three '1' and concatenate these words lexicographically ordered to obtain the word w_{n-3}. To conclude consider the infinite word

$$w = w_0 w_1 w_2 \ldots = \underbrace{111}_{w_0}\underbrace{0111\,1011\,1101\,1110}_{w_1}\underbrace{00111\,01011\,\ldots}_{w_2}\ldots$$

By construction, it is obvious that for all positive constants C, there exists n_0 such that $\forall n \geq n_0 : p_w(n) > Cn^2$. Thus w cannot be generated by a substitution and the corresponding subset W such that $\chi_W = w$,

$$W = \{0, 1, 2, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15, 16, 17, 21, 22, 23, 25, 27, 28, \ldots\},$$

is never S-recognizable.

Proposition 17 *For any numeration system S, $\mathcal{P}$ is not S-recognizable.*

Proof. In [11, 12], C. Mauduit shows using some density arguments that $\chi_{\mathcal{P}} \in \{0,1\}^\omega$ is not generated by a substitution (φ, h, α) where h sends all the letters on 0 except one. A slight adaptation of the proof leads to the conclusion for any letter-to-letter morphism h. $\square$

6 Acknowledgments

The author would like to warmly thank J.-P. Allouche for pointing out very useful references and for some fruitful comments and also P. Lecomte for his support and for fruitful conversations.

References

[1] J.-P. Allouche, Sur la complexité des suites infinies, *Bull. Belg. Math. Soc.* **1** (1994) 133-143.

[2] J.-P. Allouche, q-regular sequences and other generalizations of q-automatic sequences, *Lect. Notes in Comp. Sci.* **583** (1992) 15-23

[3] V. Bruyère, G. Hansel, Bertrand numeration systems and recognizability, *Theoret. Comput. Sci.* **181** (1997) 17-43.

[4] J. Cassaigne, Complexité et facteurs spéciaux, *Bull. Belg. Math. Soc.* **4** (1997) 67-88.

[5] F. M. Dekking, M. Mendès France, A. van der Poorten, Folds!, *Math. Intelligencer* **4** (1982), 130-138, 173-181, 190-195.

[6] A. Cobham, On the Hartmanis-Stearns problem for a class of tag machines, *IEEE Conference Record of 1968 Ninth Annual Symposium on Switching and Automata Theory*, 51-60.

[7] A. Cobham, Uniform tag sequences, *Math. Systems Theory* **6** (1972) 186-192.

[8] A. Ehrenfeucht, K. P. Lee, G. Rozenberg, Subword complexity of various classes of deterministic developmental languages without interactions, *Theoret. Comp. Sci.* **1** (1975) 59-75.

[9] P. B. A. Lecomte, M. Rigo, Numeration systems on a regular language, *preprint* (1999), see also `http://xxx.lanl.gov/abs/cs.OH/9903005`.

[10] A. Maes, Morphic predicates and applications to the decidability of arithmetic theories, *Thesis*, Université de Mons-Hainaut (1999).

[11] C. Mauduit, Sur l'ensemble normal des substitutions de longueur quelconque, *J. Number Theory* **29** No 3 (1988) 235-250.

[12] C. Mauduit, Propriétés arithmétiques des substitutions, *Séminaire de Théorie des Nombres, Paris,* (1989-90) 177-190, *Progr. Math.* **102**, Birkhäuser Boston, Boston, MA, 1992.

[13] J.-J. Pansiot, Complexité des facteurs des mots infinis engendrés par morphismes itérés, *Lect. Notes Comp. Sci.* **172** (1984) 380-389.

[14] J. Shallit, A generalization of automatic sequences, *Theoret. Comp. Sci.* **61** (1988) 1-16.